

SIMULATING METHOD FOR ROUTE ON DIGITAL MAP

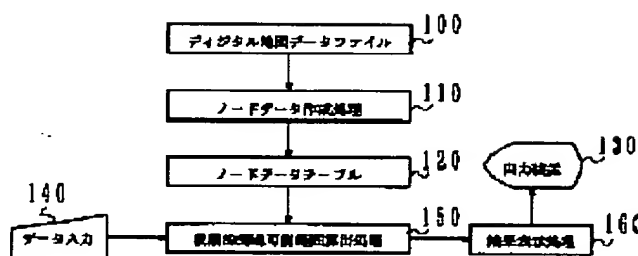
Patent number: JP8166939
Publication date: 1996-06-25
Inventor: HAGIWARA YOSHIHIRO; SETO YOICHI; HAMADA CHIGUSA; KITAZAWA SHUJI
Applicant: HITACHI LTD
Classification:
 - international: **G06F17/00; G06F17/30; G06F19/00; G09B29/10; G06F17/00; G06F17/30; G06F19/00; G09B29/10; (IPC1-7): G06F17/00; G09B29/10**
 - european:
Application number: JP19940310456 19941214
Priority number(s): JP19940310456 19941214

Report a data error here

Abstract of JP8166939

PURPOSE: To speed up the route simulation on a digital map and to improve a value added by using a highly precise Dijkstra's algorithm on the periphery of a starting point and main roads and using a high speed priority link method at a peripheral part where the number of links is large and branch roads.

CONSTITUTION: A deliverer inputs a present position and delivery scheduled time to a delivery possible range decision supporting system against the roads and a main delivery place, which are displayed on a screen. Thus, a range which the deliverer can deliver within driving scheduled time is displayed and the main delivery possible place in an area is displayed. In such a case, a node data generation processing 110 takes the lengths of the links almost equal, and the link is divided when the deviation of the link is large so as to arrange it to the length of the other links. A stepwise arrival possible range calculation processing 150 realizes high speed and highly precise retrieval by combining the high speed search method and a highly precise retrieval method. Namely, the main roads are calculated by using the highly precise Dijkstra's algorithm and the branch roads except for the main roads are calculated by using the high speed priority link method.



Data supplied from the esp@cenet database - Worldwide

BEST AVAILABLE COPY

(19) 日本国特許庁 (JP)

(12) 公開特許公報 (A)

(11) 特許出願公開番号

特開平 8 - 1 6 6 9 3 9

(43) 公開日 平成 8 年 (1996) 6 月 25 日

(51) Int. Cl. ⁶	識別記号	庁内整理番号	F I	技術表示箇所
G 0 6 F 17/00				
G 0 9 B 29/10	A	9168-5 L	G 0 6 F 15/20	D

審査請求 未請求 請求項の数 1 0 O L (全 2 4 頁)

(21) 出願番号 特願平 6-310456

(22) 出願日 平成 6 年 (1994) 12 月 14 日

(71) 出願人 000005108

株式会社日立製作所
東京都千代田区神田駿河台四丁目 6 番地

(72) 発明者 萩原 義裕

神奈川県川崎市麻生区王禅寺 1099 番地 株
式会社日立製作所システム開発研究所内

(72) 発明者 瀬戸 洋一

神奈川県川崎市麻生区王禅寺 1099 番地 株
式会社日立製作所システム開発研究所内

(72) 発明者 浜田 ちぐさ

神奈川県川崎市麻生区王禅寺 1099 番地 株
式会社日立製作所システム開発研究所内

(74) 代理人 弁理士 磯村 雅俊

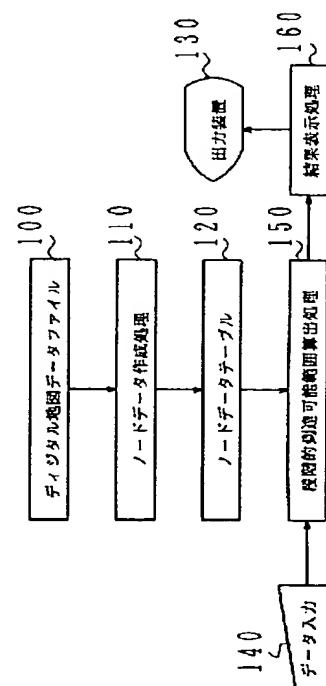
最終頁に続く

(54) 【発明の名称】 デジタル地図の経路シミュレーション方法

(57) 【要約】

【目的】 デジタル地図における経路シミュレーションの高速化と高付加価値化を実現する。例えば、段階的計算法や逆経路算出法により高速化を達成でき、ソート済みノードテーブルや、道路状況を推定処理により高精度化を達成できる。

【構成】 リンク通過時間を短い順にソートするノードデータ作成処理部と、その結果を格納するノードデータテーブルと、地図情報を格納するデジタル地図データファイルと、出発点を入力するデータ入力処理部と、ダイクストラ法と優先リンク法からなる段階的到達可能範囲の算出処理部と、シミュレーション結果を表示する結果表示処理部より構成する。



【特許請求の範囲】

【請求項 1】ファイルに格納されたデジタル地図データを基に、ノード間のリンク通過時間を算出するノードデータ作成処理と、

入力装置から出発点を入力するデータ入力処理と、
上記リンク通過時間を格納したノードデータテーブルを用いて、上記出発点から到達可能な範囲を算出する到達可能範囲算出処理と、

該到達可能範囲を出力する処理とを有するデジタル地図の経路シミュレーション方法であって、

上記ノードデータ作成処理では、リンク長の偏差が大きいとき、長いリンクの途中で仮想ノードを設け、複数の短いリンクに分割する処理を行い、また上記到達可能範囲算出処理では、主要幹線道路の到達可能範囲算出には高精度探索方法を用い、

該主要幹線道路から派生する枝道になるに従って高速探索方法に移行する連続移行型到達可能範囲算出処理を行うか、あるいは、

該主要幹線道路から派生する枝道になった時点で高速探索方法に切り替える段階的到達可能範囲算出処理を行うことを特徴とするデジタル地図の経路シミュレーション方法。

【請求項 2】請求項 1 に記載のデジタル地図の経路シミュレーション方法において、前記到達可能範囲算出処理では、出発点からの初期移動方向以外の経路のリンク通過時間及びノード通過時間を大きくするか、あるいは棄却することにより、到達可能範囲を絞り込む到達点絞り込み処理を行うことを特徴とするデジタル地図の経路シミュレーション方法。

【請求項 3】請求項 1 または 2 に記載のデジタル地図の経路シミュレーション方法において、前記到達可能範囲出力処理では、行政界ごとに異なる色で到達可能範囲を表示することを特徴とするデジタル地図の経路シミュレーション方法。

【請求項 4】ファイルに格納されたデジタル地図データを基に、ノード間のリンク通過時間を算出するノードデータ作成処理と、

入力装置から出発点を入力するデータ入力処理と、
上記リンク通過時間を格納したノードデータテーブルを用いて、上記出発点から到達可能な範囲を算出する到達可能範囲算出処理と、

該到達可能範囲を出力する処理と、

該到達可能範囲算出処理の結果を利用した逆経路出力処理とを有することを特徴とするデジタル地図の経路シミュレーション方法。

【請求項 5】請求項 4 に記載のデジタル地図の経路シミュレーション方法において、前記逆経路算出処理では、往路と復路の通過時間が等しいときは段階的到達可能範囲算出処理の結果を用い、異なるときは復路のリンク通過時間による到達可能範囲算出処理の結果を用いて

最短の逆経路を出力することを特徴とするデジタル地図の経路シミュレーション方法。

【請求項 6】請求項 4 または 5 に記載のデジタル地図の経路シミュレーション方法において、前記到達可能範囲算出処理と前記逆経路算出処理では、ノード番号を元に算出結果を検索することを特徴とするデジタル地図の経路シミュレーション方法。

【請求項 7】ファイルに格納されたデジタル地図データと、現実の道路状況を入力する道路状況センサとを基に、ノード間のリンク通過時間を算出するノードデータ作成処理と、

入力装置から出発点を入力するデータ入力処理と、
該リンク通過時間を格納するノードデータテーブルを用いて、上記出発点から到達可能な範囲を算出する到達可能範囲算出処理と、

該到達可能範囲を出力する処理と、

隣接する道路状況と周辺的主要幹線道路の道状況に応じて、各道路の通過時間を設定する道路状況設定処理とを有することを特徴とするデジタル地図の経路シミュレーション方法。

【請求項 8】請求項 7 に記載のデジタル地図の経路シミュレーション方法において、前記道路状況設定処理は、道路状況が既知の道路の車両通行密度と平均速度から隣接する道路の道路状況を推定するルールを設け、該ルールにより未知の道路状況を推定する道路状況推定処理を含むことを特徴とするデジタル地図の経路シミュレーション方法。

【請求項 9】請求項 7 または 8 に記載のデジタル地図の経路シミュレーション方法において、前記道路状況設定処理は、相互に隣接する道路の道路状況が矛盾しないように再設定を行う矛盾相殺処理を含むことを特徴とするデジタル地図の経路シミュレーション方法。

【請求項 10】請求項 9 に記載のデジタル地図の経路シミュレーション方法において、前記矛盾相殺処理は、計算済み隣接道路状況と道路同士の相関関係を格納する相関テーブルから、道路状況推定対象である道路の道路状況を推定する処理を繰り返し行うことで、確率的緩和法により矛盾を相殺することを特徴とするデジタル地図の経路シミュレーション方法。

【発明の詳細な説明】

【0001】

【産業上の利用分野】本発明は、デジタル計算機を用いたデジタル地図における経路シミュレーション方法に関し、特に宅配店において一定時間内に配達できる範囲を決定する方法、あるいはタクシーによる配車決定支援方法等において、ノード（頂点、交差点）数、リンク（隣接するノード同士を直接結ぶ短い経路、交差点間の道路）数、経路数（あるノードから任意のノードを結ぶリンクの集合）が多いためにシミュレーション時間がかかる場合や、道路状況が時間や場所の条件により変動す

る場合に好適なデジタル地図の経路シミュレーション方法に関する。

【0002】

【従来の技術】従来、デジタル地図上の経路シミュレーション方法としては、例えば、石畑清著「アルゴリズムとデータ構造」（岩波書店、1989年発行）の第4章に記載されたダイクストラ法が提案されている。この方法は、各ノードの最短経路を出発点に近いところから1つずつ確定していく方法であって、極めて正確であるが処理時間がかかり過ぎるという問題がある。なお、デジタル地図とは、デジタル計算機を使用して描く地図のことを意味する。以下、上記ダイクストラ法を説明する。図27は、ダイクストラ法の説明図である。

(1) 初期化

図27に示すように、ノード n_0, n_1, n_3, n_5 、およびノード n_0, n_1, n_4, n_6 、およびノード n_0, n_2, n_7 、およびノード n_0, n_2, n_8 の各経路があると仮定する。いま、到達時間が確定したノードの集合を V 、未確定ノードの集合を U 、到達時間計算中の確定候補ノードの集合を W として、図27に示すように、 V, W, U の各レジスタを設定する。集合 U に含まれる各ノードには、そのノード（以下、該当ノード）から該当ノードに接続するノード（以下、隣接ノード）への移動に要する時間（以下、リンク通過時間）を与えておく。まず、 U, V, W を全て空集合にする（各レジスタを空にする）。次に、以下の処理を行う。確定済みノードの集合 V に出発点 n_0 を加える（ V レジスタに n_0 格納）。確定候補ノードの集合 W に n_0 の隣接ノードを全て加える（ W レジスタに n_1, n_2 を格納）。それ以外のノードを全て未確定ノードの集合 U に加える（ U レジスタに $n_3 \sim n_8$ 格納）。また、出発点からの到達時間を表す変数 $d[n_0]$ を0に設定する（ d は到達時間を配列したレジスタ）。さらに、出発点以外の全てのノードについては、 $d[n_i]$ を無限大（あらゆる計算結果より大きい値）に設定する。

【0003】(2) 確定候補ノード W が空集合でない間、次の処理を繰り返す。

(2-1) W の要素で到達時間が最も小さいノード p を W から除き V に加える（出発点 n_0 から n_1 までの距離と n_0 から n_2 までの距離を比較し、 n_1 の方が小さいため、 W から除き（ \times ）、 V に格納する（矢印））。

(2-2) p の全隣接ノード x について次の処理を行う。もし、 x が U に属しているならば、 x を U から除き W に加える（ n_1 の隣接ノード n_3, n_4 を U から除き（ \times ）、 W に格納する（矢印））。

それまで求めた到達時間より p を経由して至る経路の到達時間が小さいとき、 x の到達時間を書き換える（ n_2 までと、 n_1 を経由して n_3 までと、 n_4 までの各距離を比べ、 n_2 が最も小さく、次に n_3 の方が小さいので、 n_2, n_3, n_4 を W から V に移して格納した後、 d レジスタ内の n_1

を求めた距離に書き替え、次に n_2, n_3, n_4 を書き替える。）。同じようにして、 n_2 を経由して n_7, n_8 までの距離と n_1, n_3 を経由して n_5 、または n_1, n_4 を経由して n_6 までの距離を比べ、 n_8, n_7 の順に小さく、次いで n_5, n_6 の順であるから、 U レジスタ内の n_8, n_7, n_5, n_6 を除き、この順序で W レジスタに格納する。次に、これらの n_8, n_7, n_5, n_6 を W から V に移し格納した後、 d の値を書き替える。要するに、ダイクストラ法はデジタル地図における道路をリンク、交差点をノードに見立て、各ノードの最短経路を出発点に近いところから1つずつ確定していく方法である。なお、この場合に、 n_2 と n_4 を結ぶリンクが存在するか否かを判断する必要があり、存在するときには n_2 を経由して n_7, n_8 に到達する距離と、 n_2 を経由して n_4, n_6 に到達する距離も比較する。

【0004】次に、ダイクストラ法から容易に類推できる方法として、ノードを順次探索する優先リンク法がある。優先リンク法の概略を以下に記す。図28は、優先リンク法の説明図である。

(1) 初期化

まず、 U, V, W を全て空集合にする。次に以下の処理を行う。確定済みノードの集合 V に出発点 n_0 を加える。確定候補ノードの集合 W に n_0 の隣接ノードを全て、リンク通過時間が短い順に加える（ n_1, n_3, n_2 を W に格納）。それ以外のノードを全て未確定ノードの集合 U に加える（ n_4 を U に格納）。また、出発点からの到達時間を表す変数 $d[n_0]$ を0に設定する。さらに、出発点以外の全てのノードについて $d[n_i]$ を無限大に設定する。ダイクストラ法と異なる点は、ダイクストラ法では出発点に近いところから最短経路を1つずつ確定して V に格納していくが、優先リンク法では、 W に加える際にリンク通過時間の短い順序で加えていき、その順序で V に加えていくので、迅速処理が可能である。

(2) 確定候補ノード W が空集合でない間、次の処理を繰り返す。

(2-1) W の要素で、一番初めに加えたノード p を W から除き V に加える。

(2-2) p の全隣接ノード x について、リンク通過時間が小さい順に次の処理を行う。もし、 x が U に属しているならば、 x を U から除き W に加える。 x の到達時間を書き込む。計算を誤った場合、例えば、図28において、 W には出発点 n_0 から近い順に n_1, n_3 を加える際には正確な順序で加えるが、次に n_3 を経由して n_4 に到達する距離の方が n_2 に到達する距離よりも小さいと誤判断して W に加えるおそれがある。要するに、優先リンク法はノードを W に加えた順に確定していく方法である。優先リンク法とダイクストラ法の相違は、手順

(2-1) で W から V に移す要素を決定する方法にある。ダイクストラ法が W の要素を全て調べてから決定するの

に対し、優先リンク法は要素をWに加えた順にVに移す。そのため、到達時間には誤差を生じるが、高速な算出が可能になる。

【0005】さらに、地図に対して実的な方法である段階的探索方法を説明する。上記方法（優先リンク法）でも処理時間が要求を満たさない場合、段階的な探索を行って見かけ上の処理時間を高速化する方法がある。公知の段階的探索方法を以下に説明する。

(1) 先ず、高速道路、国道など主要幹線道路の探索を行い、到達可能範囲を概算し、表示する。

(2) 次に、県道など準主要幹線道路を含めた探索を行い、到達可能範囲を概算し、表示する。

(3) 最後に、枝道などを含めた全道路の探索を行い、到達可能範囲を概算し、表示する。

このように、公知の段階的探索法は、暫定結果を適時表示できるため、結果を表示するまでの時間が見かけ上短くなる。

【0006】

【発明が解決しようとする課題】前述のように、ダイクストラ法や優先リンク法により、デジタル地図上である地点を出発点として設定時間内における到達可能範囲を求める場合、次のような課題がある。

(1) ダイクストラ法を用いると、ノードの数が多くなるほど処理時間がかかるという問題がある。また、優先リンク法では、経路が複数存在する場合、最短経路でない方を先に計算してしまうと誤差を生じる。誤差は経路長に比例するため、シミュレーション範囲が拡大するほど、最短経路探索誤差が大きくなるという問題がある。公知の段階的探索法では見かけ上計算処理が速くなるが、誤差は軽減できない上、全道路の探索が終了するまでの処理時間は同じか、逆に長くなるという問題がある。特に、出発点付近や主要幹線道路での誤差が生じた場合、到達可能範囲の周辺部や枝道に波及し、大きな誤差要因となる。また、周辺部や枝道は大きな誤差要因にはならないが、リンク数が多いため処理時間増加の要因となる。

(2) 優先リンク法ではノード数が少ない経路を最短とみなすため、リンク通過時間の偏差が大きい場合には、通過ノード数が少なく経路通過時間が長い経路を先に計算してしまい誤差を生じることがある。

(3) 出発点からの大まかな移動方向が既知の場合、実際の到達可能範囲は移動方向以外の方向にはないことが考えられる。従来方法では、移動方向が考慮されていないため、実際の到達可能範囲より広い範囲が算出されるという問題がある。

【0007】(4) 行政界(県や市町村の境界)にまたがる到達範囲を表示する場合、単色の表示では利用者が行政界を誤認する可能性がある(例えば、消防車や救急車の行政境界が困難)。

(5) 例えば、ピザの配達やタクシーの配車の場合、帰

路として到達可能範囲内から原点に戻るためには、最短な逆経路を求める必要がある。最短逆経路を求めるとき、往路と復路のリンク通過時間が等しいとき、到達可能範囲算出と最短逆経路算出を独立して行うと処理時間がかかるという問題がある。また、往路と復路のリンク通過時間が異なるとき、到達可能範囲算出結果の往路経路は復路の最短逆経路と等しくないため、到達可能範囲算出と最短逆経路算出を独立して行う必要がある。このとき、複数の到達点からの経路算出処理を、到達点の個数分繰り返す行わなければならないため、処理時間がかかるという問題がある。

(6) 正確な経路シミュレーションを行うためには、リンク通過時間を、その道路の道路状況に応じて設定する必要がある。その道路の道路状況が未知の場合、隣接する道路の道路状況や周辺の主要幹線道路の道路状況を用いて推定する必要がある。リンク通過時間が実際の値とずれている場合には、ダイクストラ法や優先リンク法では誤差を生ずる。

本発明の目的は、これら従来の課題を解決し、デジタル地図の経路シミュレーションの高速化と高付加価値化を実現できるデジタル地図の経路シミュレーション方法を提供することにある。さらに詳細には、段階的経路シミュレーション方法や、到達可能範囲算出結果を利用した逆経路算出方法により、精度を劣化させずに高速化を可能にし、またリンク精度をリンク通過時間が小さい順序でソートしたデータテーブルや、実測値が得られる主要幹線道路の道路状況に応じて周囲の枝道の道路状況を推定する方法により、速度を劣化させずに高精度化を可能にする。

【0008】

【課題を解決するための手段】上記従来技術の課題や問題点を解決するため、本発明では、以下の点を重要な特徴点とした。

(1) ノード数が比較的少なく精度を要する、出発点周辺や主要幹線道路の経路算出に高精度なダイクストラ法を用い、ノード数は多いが精度を要しない、到達可能範囲の周辺部や枝道の経路算出に高速な優先リンク法を用いる。このとき、ダイクストラ法と優先リンク法を切り換える方法と、ダイクストラ法から優先リンク法に次第に移行する方法があるので、いずれか一方を用いる。

(2) リンク通過時間の偏差が大きい場合には、長いリンクの途中に仮想ノードを設け、複数の短いリンクに分割し、全てのリンクの通過時間をおおよそ揃える。

(3) 出発点からの大まかな移動方向が既知の場合、移動方向以外の経路の経路通過時間を大きくする。経路通過時間を大きくする方法には、移動方向以外にあるリンク通過時間を大きくする方法、移動方向以外にあるノードを通過するための所要時間(ノード通過時間)を大きくする方法、移動方向と異なる向きのリンクのリンク通過時間を大きくする方法がある。また、到達可能範囲算

出後、移動方向以外の経路を棄却する方法もある。これらのうちの一つを用いればよい。

(4) 結果表示の際、行政区ごとに異なる色、線種、タイルパターンを用いて表示を行う。なお、タイルパターンとは、各区画毎に種々の模様を使用したパターンのことである。

(5) 到達可能範囲内から原点に戻るための最短逆経路計算の際、往路と復路のリンク通過時間が等しいときは、到達可能範囲算出に伴って算出される最短往路を最短逆経路として最短逆経路を出力する。往路と復路のリンク通過時間が異なるときは、逆経路のリンク通過時間を用いて出発点からの到達可能範囲を計算することにより、到達可能範囲内の任意のノードからの最短逆経路を算出する。

(6) リンク通過時間の実測値が得られる主要幹線道路の道路状況に応じて、周囲の枝道のリンク通過時間を推定する。また、実測値及び推定値をもとにリンク通過時間を更新する。

【0009】

【作用】上記各特徴点は、おのおの以下の作用がある。

(1) 出発点付近や主要幹線道路に高精度なダイクストラ法を用いることにより、周辺部や枝道に波及する誤差要因はなくなる。これらのリンク数が占める割合は比較的少ないため処理時間の増加は小さい。また、リンク数が多い周辺部や枝道では、高速な優先リンク法を用いることにより、到達可能範囲算出に要する時間の増加要因はなくなる。この部分で生じる誤差は他の部分に波及しないため、到達範囲算出誤差の増加を防ぐことができる。これにより、経路誤差の増加を抑えて計算時間の短縮が可能になる。

(2) 長いリンクの途中で仮想ノードを設け、複数の短いリンクに分割することにより、リンク通過時間の偏差を低減することができる。リンク通過時間の偏差が小さくなると、通過ノード数が少なくリンク通過時間が長い経路は減少する。これにより、優先リンク法を利用した場合の誤差を少なくすることができる。

(3) 移動方向以外の経路上のリンク通過時間やノード通過時間を大きくすることにより、その方向への経路通過時間が大きくなり、到達範囲は狭くなる。また、到達可能範囲算出後、移動方向以外の経路を棄却することにより、移動方向と同じ方向へ向かう経路のみが残る。これらにより、初期の移動方向を加味し、実際の到達可能範囲に即した結果を得ることができる。

(4) 行政区ごとに色を変えることにより、表示結果の視認性が向上する。これにより、利用者が行政区を誤認する可能性を低減することができる。

(5) 到達可能範囲算出結果を用いて最短逆経路を出力することにより、到達可能範囲内から原点に戻るための最短逆経路を高速に求めることができる。これにより、経路シミュレーションの計算時間短縮が可能になる。

(6) 主要幹線道路の道路状況に応じて周囲の枝道の道路状況を推定し、逐次更新することにより、センサが設置されていない道路の通過時間を実際の値に近づけることができる。これにより、経路シミュレーションの高精度化が可能になる。

【0010】

【実施例】

〈第1の実施例〉以下、本発明の実施例を、図面に基いて詳細に述べる。図1は、本発明の第1の実施例を示す配達可能範囲決定支援システムの機能ブロック図であり、図2は一例としてカーナビゲーションシステムの表示画面を用いて、配達可能範囲決定システムの機能を説明する図である。本実施例のシステムは、例えば宅配店において一定時間内に配達できる範囲の決定を支援するシステムであって、図1～図20により説明する。本システムは、ディスプレイに表示されている地図に対して、現在地と配達時間を入力することにより、予定時間内に配達できる範囲を決定するものである。道路を通過するために要する時間は渋滞などにより、刻々と変化するため、主要幹線道路の道路状況をセンサにより入力する。これらの情報とデジタル地図を基にあらかじめ道路ノードテーブルを作成し、到達可能範囲算出に用いる。本システムの機能の概略を、図2により説明する。画面上に表示された道路260（図中点線）と主要配達地点200等（図中○印）に対し、配達者が現在位置（図中●印）230と配達予定時間を配達可能範囲決定支援システムへ入力することにより、運転予定時間内で配達可能な範囲（到達可能道路220）を図中実線のように表示し、領域内（図中実線内）の主要配達可能地点240等（図中二重丸印）を表示する。図中の太線は、ある主要幹線道路である。この場合、ノードデータ作成処理110では、リンクの長さ（交差点相互間の距離）をほぼ等しくとり、リンクの偏差が大のときにはリンクを分割して、他のリンクと長さを揃える。また、段階的到達可能範囲算出処理150では、高速探索法と高精度探索法とを組み合わせ、高速高精度な探索を実現する。例えば、太線の主要幹線道路は高精度のダイクストラ法を利用して算出し、それ以外の枝道では高速度の優先リンク法を利用して算出する。なお、出発点周辺を高精度に、周辺部を高速度に、探索することもできる。さらに、図2において、出発点230から配達可能地点210に配達する場合に、両地点に直結する道路に平行なリンクに対してはリンク通過時間を最短時間で算出し、それ以外のリンクに対しては通過時間を大きくするか、あるいは両地点に直結する道路に垂直なリンクに対してはリンク通過時間を最大にし、それ以外のリンクに対しては短時間で算出するようにしてもよい。

【0011】以下、図1により本実施例における処理の詳細を述べる。ノードデータ作成処理110と段階的到達可能範囲算出処理150が計算機内の処理であり、ディジ

タル地図データファイル100とノードデータテーブル120は計算機内のメモリ内に格納されるファイルおよびテーブルである。データ入力140、出力装置130は入出力装置であり、結果表示処理160は入出力制御装置内および計算機内での処理である。図3は、図1におけるノードデータ作成処理部の機能ブロック図であり、図4は、ノードデータ作成処理部の動作フローチャートである。

1. ノードデータ作成処理110

デジタル地図データファイル100より、道路ノードデータテーブル120を作成する。ノードデータ作成処理110は、図3に示すような機能構成であり、図4のフローにより実行される。

【0012】 (1) 主要幹線道路状況データテーブル作成処理315

図5は、図3における主要幹線道路状況データテーブルの様式を示す図であり、図6は、図3におけるデジタル地図データファイルの様式を示す図である。刻々と変わる道路の混雑状況や平均車両速度を、道路情報335とする。そのために、主要な幹線道路に道路情報335を計測するための主要幹線道路状況センサ310を設置する。主要幹線道路状況データテーブル作成処理315は、主要幹線道路状況センサ310より入力された主要幹線道路情報335とデジタル地図データファイル100を基に、主要幹線道路状況データテーブル340を作成する。主要幹線道路状況データテーブル340は、図5に示すような構成である。このうち、主要幹線のノード番号510、隣接ノード数520及びノードポインタ530を、デジタル地図データファイル100より複写する。デジタル地図データファイル100には、図6に示すようにX座標600、Y座標610、隣接ノード数620、隣接ノード番号640、主要幹線道路フラグ630、リンク長650が格納されている。主要幹線道路フラグ630には、隣接ノードへのリンク上にセンサが設置されているとき(1で示す)、センサが設置されていない主要幹線道路のとき(2で示す)、枝道のとき(0で示す)が格納されている。このフラグ630を利用して、フラグ2のときは高精度探索を、フラグ0のときは高速度探索を行うことができる。

【0013】 図7は、本発明を説明するための交差点および道路模式図である。主要幹線道路センサ310は、例えば路上に設置した車両センサであり、図2における道路250のうち、主要幹線道路上を通行する車両の密度と平均速度を測定し、図5における車両通行密度550と車両平均速度540に保存する。例えば、図7において、ノード(49)からノード(51)への道路上にセンサが設置されている場合、以下のような処理を行う。このとき $k=0$ とする。まず、デジタル地図データファイル上のノード(49)のデータより、隣接ノード番号が51であるものを探す。この例では $i=49$ 、 $j=2$ である。この場合、 i は図6の欄外の $51 \sim i$ の値であり、49は最上段に記入されているべき値であり、また j は主要幹線道路

フラグの2(センサが設置されていない)である。そこで、主要道路状況テーブルに、ノード番号(49)、隣接ノードポインタ(j)を書き込む。隣接ノードポインタには、2が書き込まれる。最後に、センサから入力した平均速度、車両通行密度を書き込み、 k に1を加える。以上の処理をセンサの数だけ繰り返すことにより、主要幹線道路状況データテーブル340を作成する。

【0014】 (2) 道路状況算出320

図8は、図3における道路状況データテーブルの様式を示す図である。道路状況データテーブル345の初期値を作成する。道路状況の算出には道路状況ルールデータテーブル300に格納されたルールとデジタル地図データファイル100、主要幹線道路状況データテーブル340に格納されたデータを用い道路状況算出処理320により行う。図4において、先ず初めに主要幹線道路状況データテーブル340の初期化310を行う(ステップ410)。すなわち、道路状況データテーブル345の車両通行密度840と車両平均速度830を0に設定する。次に、主要幹線道路状況データテーブル340の内容を道路状況データテーブル345に転送420し、決定フラグを1にする(ステップ420)。未決定のリンクは決定フラグを0にしておく。道路状況データテーブル345上の未計算リンク(決定フラグが0、すなわち、車両通行密度840と車両平均速度830が未計算のもの)に隣接するリンクが設定済みの場合(ステップ430)、未計算リンク通過時間を計算する(ステップ440)。この処理を、未計算リンクがなくなるまで繰り返し行う(ステップ450)。道路状況設定ルールテーブル300には、例えば次のようなルールが格納されている。かつこ内は変数、 $func$ は関数である。

【0015】 ルール：ノード($x0$)からノード($x1$)へのリンク速度 $qv[x0][x1]$ が既知で、他の既知リンクが($m[x1]$)本るとき、ノード($x1$)からノード($x2$)へのリンク速度 $qv[(x1)][(x2)]$ は($func((qv[x1][x2])(qv[x0][x1]), m[x1])$)である。例えば、下記のようなルールを、道路条件ルールデータテーブル300に格納する。

具体的なルール：ノード(50)からノード(51)へのリンク速度 $dv[50][51]$ が既知で、他の既知リンクが(2)本るとき、ノード(51)からノード(52)へのリンク速度 $dv[(51)][(52)]$ は($dv[50][51] + dv[50][51]/2$)である。

【0016】 (3) 矛盾相殺処理325

図9および図10は、図3における隣接道路相関係数テーブルの様式を示す図である。ここでは、道路状況データテーブル345における矛盾を相殺する。矛盾の相殺は、道路状況算出処理320で作られた道路状況データテーブル345と隣接道路相関テーブル305を用い、矛盾相殺処理325において行う。計算中に矛盾が生じた場合、例えばある交差点の入り口には車両が大量に流れ込んでいるのに、出ていく車両がないときなどでは、確率的緩

和法により矛盾を解消する。確率的緩和法は、周囲の状況から注目地点の状況を推定する手法である。注目地点の状況は周囲のある地点が注目地点に対して影響を与える度合いと、その地点の状況の積和によって求める。

(この詳細は、例えば、SPIDER作業グループ編「SPIDER USERSMANUAL」(SPIDER作業グループ、1989年発行)参照されたい)。前記の例の場合は、確定済みのリンクに対し変更を加える(ステップ460)。すなわち、交差点に流入するリンクの車両数を減らし、流出するリンクの車両数を増やす。ノードx1からノードx2に接続するリンクの道路状況を、隣接道路関連テーブル305と道路状況データテーブル345を用いて計算する。ノードx0からノードx1に接続するリンクの車両速度と、ノードx1からノードx2に接続するリンクの関係の強さを、あらかじめ隣接道路関連テーブル上の相関係数に格納しておく。

【0017】例えば、図7において、ノード(51)と(52)が共に3叉路で、ノード(51)からノード(52)に接続するリンクに流入する車両はノード(50)とノード(49)から、また、流出する車両はノード(53)と(54)に流出するとする。このとき、ノード(51)からノード(52)に関係するリンク(関係リンク)数920は、自分自身を含めて5となる。ノード(51)の隣接ノード数bn[51]は3、隣接ノード番号bc[51][0]、bc[51][1]、bc[51][2]、にそれぞれ(49)、(50)、(52)という値を格納してあるならば、関係リンク数b1[51][2]には5が格納されている。関係リンク始点930には、関係リンクの始点ノード番号が格納されている。また、関係リンクポインタ940には、「各リンク情報が始点ノード情報の何番目に格納されているか」という情報が格納されている。例えば、bs[51][2][0]が49のときノード(49)から(51)の車両平均速度は、道路状況テーブル上のdv[49][bp[51][2][0]]である。

【0018】一方、速度←速度相関係数950において、bvv[51][2][0]には、ノード(49)と(51)を結ぶリンクの車両平均速度が、ノード(51)と(52)を結ぶリンクの車両平均速度に与える影響の大きさを格納しておく。速度←密度相関係数960において、bvm[51][2][0]には、ノード(49)と(51)を結ぶリンクの車両通行密度が、ノード(51)と(52)を結ぶリンクの車両平均速度に与える影響の大きさを格納しておく。速度←速度相関係数950において、bmm[51][2][0]には、ノード(49)と(51)を結ぶリンクの車両通行密度が、ノード(51)と(52)を結ぶリンクの車両通行密度に与える影響の大きさを格納しておく。密度←速度相関係数980において、bm v[51][2][0]には、ノード(49)と(51)を結ぶリンクの車両平均速度が、ノード(51)と(52)を結ぶリンクの車両通行密度に与える影響の大きさを格納しておく。

【0019】関係する全てのリンクの車両速度と密度、

および相互の影響の度合より、ノード(51)と(52)を結ぶリンクの道路状況を再計算する。例えば、確率的緩和法を用いた場合、このリンクの平均速度は次のよう求める。

qv[b1[51][2][0]][bp[51][2][0]]×bvv[51][2][0]
+qv[b1[51][2][1]][bp[51][2][1]]×bvv[51][2][1]
+qv[b1[51][2][2]][bp[51][2][2]]×bvv[51][2][2]
+qv[b1[51][2][3]][bp[51][2][3]]×bvv[51][2][3]
+qv[b1[51][2][4]][bp[51][2][4]]×bvv[51][2][4]
+qm[b1[51][2][0]][bp[51][2][0]]×bvm[51][2][0]
+qm[b1[51][2][1]][bp[51][2][1]]×bvm[51][2][1]
+qm[b1[51][2][2]][bp[51][2][2]]×bvm[51][2][2]
+qm[b1[51][2][3]][bp[51][2][3]]×bvm[51][2][3]
+qm[b1[51][2][4]][bp[51][2][4]]×bvm[51][2][4]

ここで、qvはリンク速度、bvvは車両平均速度、qmは車両密度、bvmはリンクの車両通行密度である。同様の方法で、車両通行密度も計算する。車両通行密度、車両平均速度共に矛盾がなくなったとき、計算を終了する。矛盾がなくなったことを確認するためには、例えば、再計算前の車両通行密度、車両平均速度を保存して再計算後の値と比較470し、両者の差が一定のしきい値を下回ったならば、次の処理に移行する。

【0020】(4) 道路ノードデータテーブル作成330

図11は、図3におけるノードデータテーブルの様式を示す図である。図3に示すように、リンク通過時間が小さい順に、ソーティング済みのノードデータテーブル350と、主要幹線道路ノードデータテーブル355を作成する。ノードデータテーブル120の作成には、デジタル地図データファイル100と道路状況データテーブル345を用いる。リンク通過時間1040は、車両平均速度840×リンク長650により求める。各ノードについて、リンク通過時間が小さい順に、隣接ノード番号とリンク通過時間をソート480する。例えば、図7において、ノード(52)に隣接するノードはノード(51)、ノード(53)、ノード(54)の3交差点である。従って、ノードテーブルの52番目の項目の隣接ノード数は3である。また、ノード(52)からのリンク通過時間が短い順に並べるとノード(54)、ノード(53)、ノード(51)となる。従って、隣接ノード番号は(54)、(53)、(51)なる値が順に格納される。

【0021】2. データ入力140

図12は、段階的到達可能範囲算出処理の機能ブロック図であり、図13は、図12における段階的到達可能範囲算出処理の動作フローチャートであり、図14～図16は、図12における段階的到達可能範囲算出結果の表示例を示す図であり、図17は、到達可能範囲テーブルの図である。図1のデータ入力140では、配達先、運転予定時間を入力する。例えばマウスを用いて地図上で位置を指定した場合、x、y座標が入力される。入力されたx、y座標に最も近いノードを道路ノードデータテ

ブルより検索し、配達先ノード i とする。別途キーボードにより運転予定時間 t を入力する。このとき、あらかじめ地図を表示しておいてもよい。その場合は、カーナビゲーションシステム等の表示装置130に、配達先を含むデジタル地図データファイル100と道路ノードデータテーブル120の内容をベクトル地図として表示する。

3. 段階的到達可能範囲算出処理150及び結果表示処理160以上の入力データを基に、運転予定範囲内で到達可能な到達可能範囲を算出し、カーナビゲーションシステム等の表示装置130に表示する。到達可能範囲の算出には、道路ノードデータテーブル120の内容を考慮する。運転予定範囲内で到達可能な到達可能範囲を算出する処理は、図12に示すように、段階的に到達可能な範囲を算出する段階的到達可能範囲算出処理150であり、図13に示すようなフローで行われる。最初に、出発点から主要幹線道路までの計算処理1100を行う。次に、主要幹線路上の到達可能範囲をダイクストラ法により計算する主要幹線道路の経路算出処理1110を行う。最後に、枝道を含めた全道路について到達可能範囲を優先リンク法により計算する枝道の経路算出処理1120を行う。図14～図16は、シミュレーションを段階的に行う過程を示したものである。図12および図13に基づき、到達可能範囲の算出70を詳述する。なお、到達可能範囲テーブルは、図17に示すようなテーブルであり、ノード番号1400、ノードのX座標1410、Y座標1420、出発点からの到達時間1430、および各ノードに到達する時に経由する隣接ノード番号（源ノードのノード番号1440）より構成される。

【0022】 (1) 主要幹線道路までの経路算出処理1100

図13に示すように、初めに、到達可能範囲可能テーブル1140上のすべてのノードの到達時間をあらゆる到達時間より大きい値にする（ステップ1205）。例えば、到達時間の最大値が99999分を越えないなら99999にセットする。次に、出発点ノード i を始点に到達可能範囲算出処理を行う（ステップ1210）。この処理は、ダイクストラ法か順次計算法によって行う。算出経過で得られる各ノードへの到達時間1430（ダイクストラ法の場合、確定済みの到達時間）を逐次到達可能テーブル1140に書き込む処理を行う（ステップ1215）。さらに、各ノードに到達する時に経由する隣接ノード（源ノード1440）番号を、到達可能テーブル1140に書き込む。例えば、図7において、ノード（48）を始点とした場合、ノード（51）の源ノード1440はノード（50）である。主要幹線道路上のノードに到達したならば、その結果を表示する（ステップ1220）。図7において、ノード（51）からノード（52）への道路が幹線道路のとき、ノード（51）に到達した時点で結果を表示し（図11の160）、次の処理に移る。

【0023】 (2) 主要幹線道路の経路算出処理1110

幹線道路に到達したならば（ステップ1220）、そのノードを始点として（ステップ1225）、ダイクストラ法により到達可能範囲算出処理を行う（ステップ1230）。ダイクストラ法による処理では、幹線道路のみについて経路算出を行う。ここで得られた各ノードへの到達時間1430と源ノード番号1440を、到達可能テーブル範囲テーブル1140に書き込む処理を行う（ステップ1235）。到達時間が配達時間 t を越えたとき、すなわち幹線道路の探索を終了したとき（ステップ1240）、結果を表示し（160）、次の処理に移る。

(3) 枝道の経路算出処理1120

出発点を始点として再設定し（ステップ1245）、順次計算法により到達可能範囲算出処理を行う。この処理は、すべての道路について行う。まず、計算対象となるノードについて、到達可能範囲の結果を算出済みかどうか調べる（ステップ1250）。算出済みの場合その結果を到達可能範囲テーブルからシミュレーションテーブルに複写し（ステップ1255）、それをシミュレーション結果とする。算出済みでない場合には、優先リンク法により出発点からの到達可能範囲算出処理を行い（ステップ1260）、その結果を到達可能範囲テーブルに書き込む処理を行う（ステップ1265）。幹線道路上のノードは、ダイクストラ法により全て算出済みなので、主要幹線道路上のノードを通る時に誤差補正が行われる。例えば、図7において、ノード（52）は幹線道路に接続しているため、出発点からの到達可能範囲算出処理（ステップ1260）は行われず、ダイクストラ法の結果がシミュレーションテーブルに複写される（ステップ1255）。ノード（54）やノード（53）の到達時間算出処理では、ノード（52）までの経路誤差は0であるため、ノード（52）までの経路誤差が0でない一般的な優先リンク法より誤差が小さい。処理終了した後（ステップ1270）、結果表示（160）を行う。

【0024】 以上が、本発明の段階的到達可能範囲算出方法の詳細実施例である。本実施例の方法では、一定時間での行動可能範囲算出により消防車や救急車がサポートできる範囲と行動時間を決定するシステム、宅配便の集配所やピザ屋、コンビニエンスストアの位置などのサポート範囲を決定するシステム、警察における配備地点の決定指揮支援システムにも利用可能である。都市近郊では、朝夕で車両の流れが変化し、サポート可能な範囲も大きく変わる。通常、朝は上り道路、夕方は下り道路が混雑する。複数の配達先が存在する場合、朝は下り方向へ配達する配達点、夕方は上り方向に配達する配達点から配送すれば、配達時間の短縮が可能となる。なお、矛盾相殺処理では、確率的緩和法以外にも、統計的緩和法（統計表により緩和する）、ニューラルネットワーク、GA（遺伝的アルゴリズム）、ルールベース処理、リンク速度平均値、あるいはリンク速度中間値を利用して緩和処理を行うことが考えられる。

【0025】 <第2の実施例> 第2の実施例は、タクシ

一などによる配車決定支援システムである。配車決定支援システムは、配車の指揮をする者がタクシーなど配車の決定を行う判断を支援するシステムである。図 1 8 は、本発明の第 2 の実施例を示す配車決定支援システムの説明図である。本実施例は、移動中のタクシーが多数ある場合、顧客に対し最短時間で直行できる車両の配車決定を支援するシステムである。本システムの機能の概略を、図 1 8 により説明する。図 1 8 は、例えばワークステーションのウインドウ画面である。指揮者は、顧客の現在位置（図中●印）1540と配車予定時間を配車決定支援システムへ入力することにより、運転予定時間内で配車可能な範囲（到達可能範囲1510）を表示（図中実線）し、領域内の車両存在地点（図中二重丸印）1550と最短時間で到達できる道路の情報（図中太線）を表示する。なお、白丸1500は、現在、車両が存在している地点であって、二重丸を含めて4台存在するが、そのうち最短時間で顧客に到達できる車両が二重丸の1550である。以下、図 1 の配達可能範囲決定システムにより、本実施例における処理の詳細を述べる。

【0026】1. ノードデータ作成処理110

まず、デジタル地図データファイルよりノードデータテーブル120を作成する。これは第 1 の実施例と同様の処理であるが、リンク通過時間が往路と復路で異なる場合には、ノードデータ作成処理110において次の処理を行う。すなわち、往路のリンク情報と復路のリンク情報を入れ換え、逆経路算出処理用ノードテーブルを作成する。逆経路算出処理用テーブルの構成は、ノードデータテーブル120とおなじである。

2. データ入力140

各車両の現在位置、運転予定時間を入力する。これも第 1 の実施例と同様の処理である。

【0027】3. 到達可能範囲の算出150

以上の入力データを基に、運転予定範囲内で到達可能な到達可能範囲を算出する。これも第 1 の実施例と同様の処理であるが、リンク通過時間が往路と復路で異なる場合には、逆経路算出処理用ノードテーブルを用いる。

4. 逆経路算出1600、および結果表示処理160

図 1 9 は、逆経路算出処理部の機能ブロック図であり、図 2 0 は、図 1 9 における逆経路算出処理部の詳細ブロック図であり、図 2 1 は、逆経路算出処理部の動作フローチャートである。到達可能範囲内の車両位置と到達可能範囲算出結果を用い、各車両の往復路を計算し表示する。第 1 の実施例と異なるのは、図 1 9 に示すように逆経路算出処理部1600が設けられている点である。逆経路算出処理部の詳細は、図 2 0 に示すように到達可能車両検索処理1700と経路算出処理1710で構成され、図 2 1 に示すようなフローで行われる。図 1 9 および図 2 0 に基づき、逆経路算出処理1600を詳述する。

【0028】(1) 到達可能車両検索処理1700

全ての車両が存在するノード番号を調べ、そのノード番

号への到達時間が運転予定時間以内のものは、到達可能車両である。到達可能車両のノード番号と到達可能車両の合計数を調べる処理を行う（ステップ1810）。なお、一台の車両について到達可能か否かを調べて経路算出処理を行う処理を繰り返す方法も考えられる。

(2) 経路算出処理1410

到達可能車両について、経路算出処理を行う。車両の存在地点ノード i1 を、逆経路探索ノード n に設定する

（ステップ1820）。次に、到達可能範囲テーブル1

140を用い、nについて、源ノードm[i1]=i2を探索1830する。源ノードi2を逆経路探索ノードnに設定し（ステップ1840）、新たな源ノードm[i2]を探索する

（ステップ1830）。この処理をnが顧客位置ノードになるまで繰り返す（ステップ1850）。例えば、図 7 において出発ノードがノード（48）で、車両がノード（53）に存在するとき、n=53である。ノード（53）の源ノードはノード（52）、ノード（52）の源ノードはノード（51）、・・・、と再帰的に求め、n=48になったら終了である。これにより、通過すべき経路の算出が行われ

る。車両が複数の場合は、上記最短経路算出処理を全車両算出済み（ステップ1860）になるまで車両の数だけ繰り返す。

【0029】5. 結果表示160

経路シミュレーションの結果を、図 1 8 に示すデジタル地図上に表示する。太線の矢印1520が算出結果である。以上が、本発明の到達可能処理結果を用いた逆経路計算の詳細実施例であるが、店舗が複数あるビザ屋などの宅配・出張サービスで、どの店舗から配達・出張を行うかを決定するための指揮支援にも利用可能である。配達指揮者は、配達先と配達予定時間を配達可能範囲決定支援システムへ入力することにより、最大移動範囲内の最も近い配達車両と、配達車両が最短時間で到達できる往路及び復路の道路情報を決定することができる。また、消防車や救急車、警察における事故発生現場への急行車両を決定する指揮支援システムにも利用可能である。このような場合、図 2 2 のようなテーブルを用い主要な到達地点（配達先や配備箇所）を出力することにより、どの配達先へ行くかなどの判断支援ができる。また、図 2 3、図 2 4、図 2 5、および図 2 6 のように、ノード番号のポインタを用いてノード情報を取り出すことにより、さらに高速化が可能となる。

【0030】図 2 2 では、予め配備箇所テーブルに配備箇所（1、2、3・・・）を用意しており、先ずシミュレーションテーブルを用いて配備箇所の検索を行う

（1）。次に、その結果を配備結果テーブルに格納する

（2）。次に、配備箇所を地図上に表示する（3）。地図は、ノードテーブルのノード番号と座標で表わされる。次に、配備決定テーブルから配備箇所テーブルに地名を移転して記入する。図 2 3、図 2 4 には、それぞれ候補ノードおよびアドレスポイントにより候補ノードま

たはシミュレーションノードを検索する場合のテーブルが示されている。いずれのテーブルにも、ノード番号と到達時間、計算状況、呼び出し元ノード番号が登録されている。図 2 5 は、ノード番号テーブルであって、注目ノードと隣接ノードが登録されている。図 2 6 は、消防車や救急車、警察における事故発生現場への急行車両を決定する指揮支援システムに応用できるパトカーの検索方法を示す図である。まず、パトカーテーブルよりパトカーのノード番号を用いて、シミュレーションテーブルには、ノード番号、到達時間、計算状況、および呼び出し元ノード番号が登録されている。次に、パトカーが決定された後、パトカーの経路を検索するため、パトカー経路テーブルを見て、該当するパトカーのノード番号によりノードテーブルを検索し、急行パトカーとパトカーの経路表示を行う。なお、ノードテーブルには、経路のノード番号とその座標が登録されている。

【0031】

【発明の効果】以上説明したように、本発明によれば、デジタル地図における経路シミュレーションの高速化と高付加価値化を実現することができる。例えば、段階的経路シミュレーション方法や、到達可能範囲算出結果を利用した逆経路算出方法により、精度を劣化させることなく高速化が可能であり、また、リンク情報をリンク通過時間が小さい順にソートしたノードデータテーブルや、実測値が得られる主要幹線道路の道路状況に応じて周囲の枝道の道路状況を推定する方法により、速度を劣化させることなく高精度化が可能である。

【図面の簡単な説明】

【図 1】本発明の第 1 の実施例を示す配達先決定指揮支援システムの機能構成図である。

【図 2】配達先決定指揮支援システムのシミュレーション表示例の図である。

【図 3】図 1 におけるノードデータ作成処理部の詳細構成図である。

【図 4】図 1 におけるノードデータ作成処理のフローチャートである。

【図 5】図 3 における主要幹線道路状況データテーブルの様式を示す図である。

【図 6】図 3 におけるデジタル地図データファイルの様式を示す図である。

【図 7】本発明を説明するための交差点および道路の様式図である。

【図 8】図 3 における道路状況データテーブルの様式を示す図である。

【図 9】図 3 における隣接道路相関係数テーブルの様式を示す一部の図である。

【図 10】同じく隣接道路相関係数テーブルの他の一部を示す様式図である。

【図 11】図 3 における道路ノードデータテーブルの様

式を示す図である。

【図 12】図 1 における段階的到達可能範囲算出処理部の構成図である。

【図 13】図 1 1 における段階的到達可能範囲算出処理の動作フローチャートである。

【図 14】段階的到達可能範囲算出結果（その 1）の表示例を示す図である。

【図 15】同じく段階的到達可能範囲算出結果（その 2）の表示例を示す図である。

10 【図 16】同じく段階的到達可能範囲算出結果（その 3）の表示例を示す図である。

【図 17】図 1 の算出処理に必要な到達可能範囲テーブルの様式を示す図である。

【図 18】本発明の第 2 の実施例を示す配車決定支援システムのシミュレーション結果表示例の図である。

【図 19】逆経路算出処理結果出力処理の構成図である。

【図 20】図 18 における逆経路算出処理部の詳細構成図である。

20 【図 21】図 18 における逆経路算出処理部のフローチャートである。

【図 22】本発明における配備箇所検索テーブル図である。

【図 23】本発明の高精度探索方法で用いるシミュレーションテーブル図である。

【図 24】本発明の高速探索方法で用いるシミュレーションテーブル図である。

【図 25】本発明の到達可能範囲算出処理結果テーブル図である。

30 【図 26】本発明の逆経路算出処理用テーブル図の一部である。

【図 27】従来におけるダイクストラ法の説明図である。

【図 28】従来における優先リンク法の説明図である。

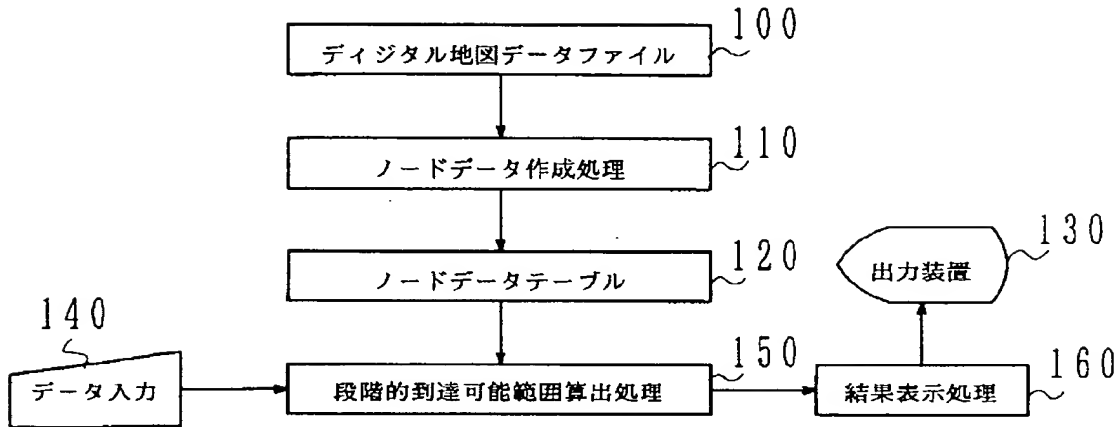
【符号の説明】

100…デジタル地図データファイル、110…ノードデータ作成処理、120…ノードデータテーブル、130…ディスプレイ、140…データ入力装置、150…段階的到達可能範囲の算出処理、160…結果表示処理、335…道路情報、310…主要幹線道路状況センサ、315…主要幹線道路状況データテーブル作成処理、320…道路状況設定処理、325…矛盾相殺処理、330…道路ノードデータテーブル作成処理、340…主要幹線道路状況データテーブル、300…道路条件ルールデータテーブル、305…隣接道路相関テーブル、345…道路状況データテーブル、120…ノードデータテーブル、350…道路ノードデータテーブル、355…主要幹線道路ノードデータテーブル、1100…主要幹線道路までの経路算出処理、1110…主要幹線道路の経路算出処理、1120…枝道の経路算出処理、1140…到達可能範囲テーブル、1130…シミュレーションテーブル、1600…逆経

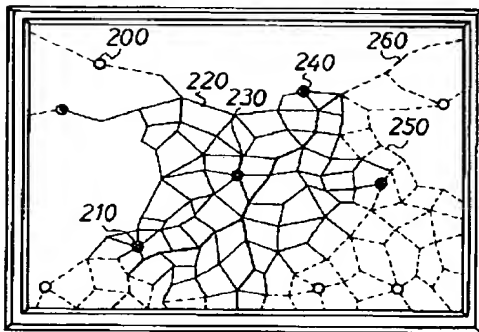
路算出処理、1700…到達可能車両検索処理、1710…経路

算出処理。

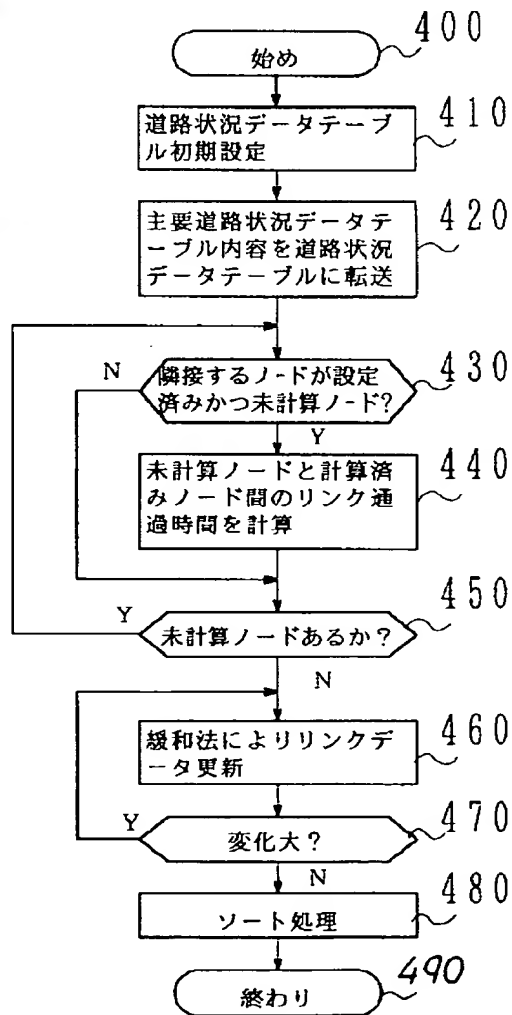
【図1】



【図2】



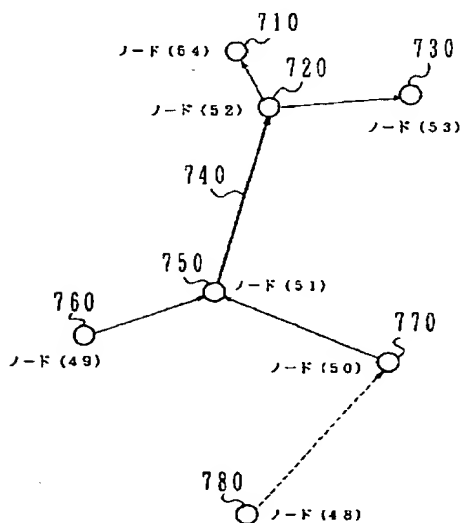
【図4】



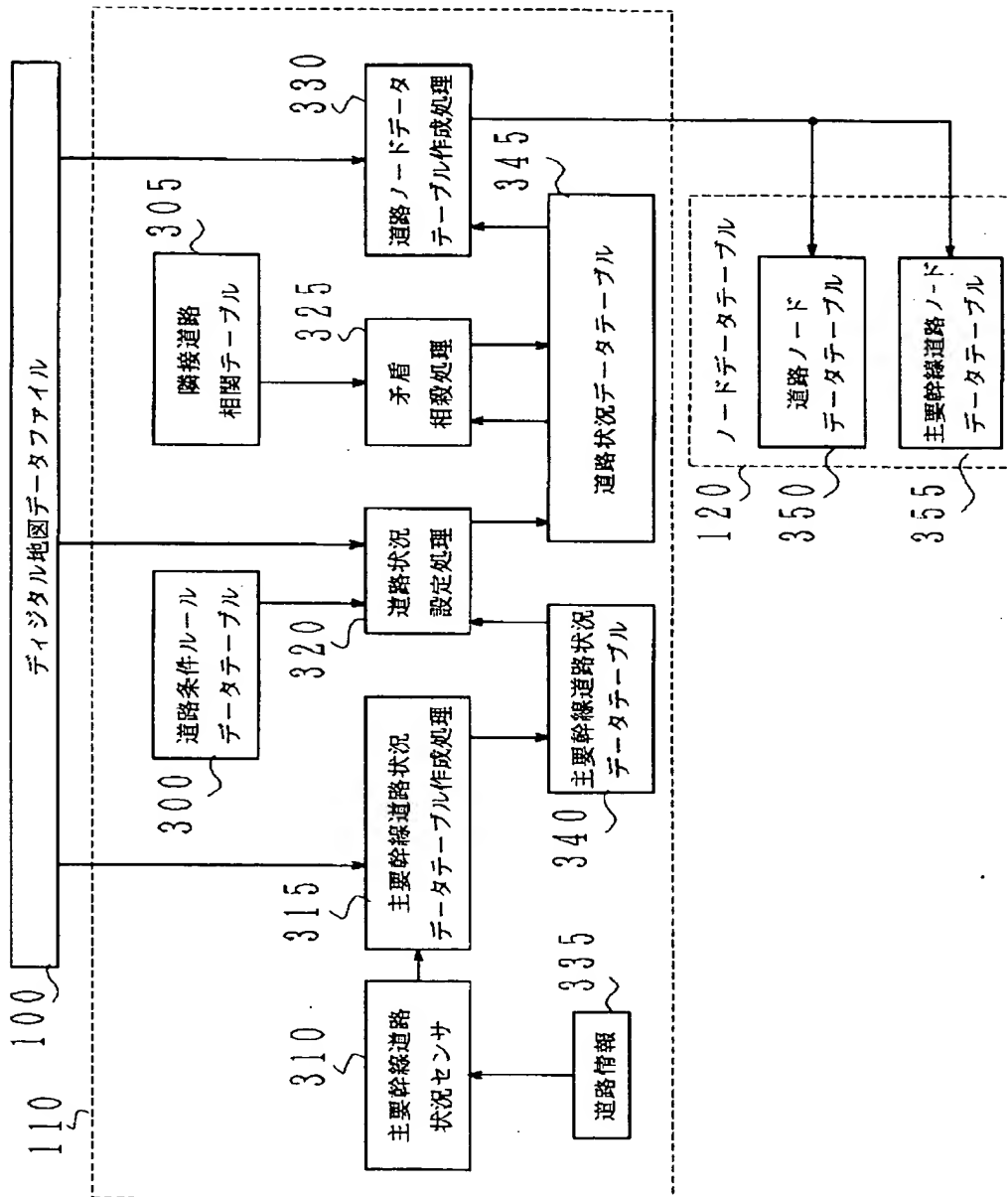
【図25】

ノード番号	
注目ノード	隣接ノード
5	7
5	3
⋮	⋮
⋮	⋮

【図7】



【図3】



【図 5】

510 ノード番号	520 隣接ノード番号	530 隣接ノードポイント	540 平均速度	550 車両密度
4 9	5 1	2	1 0	2 0
⋮	⋮	⋮	⋮	⋮
s n[k][0]	s f[k]	s c[k]	s v[k]	s b[k]

【図 8】

	800 隣接ノード数	810 決定フラグ	820 隣接ノード番号	830 平均速度	840 車両密度
	⋮	⋮	⋮	⋮	⋮
5 1	3	0	4 9	1 0	2 0
		0	5 0	1 0	1 0
		1	5 2	6 0	1 2 0
5 2	3	1	5 1	4 0	4 1 0
		0	5 3	2 0	1 2 0
		0	5 4	2 0	8 0
	⋮	⋮	⋮	⋮	⋮
i	q n[i]	q f[i][0]	q c[i][0]	q v[i][0]	q m[i][0]
		q f[i][j]	q c[i][j]	q v[i][j]	q m[i][j]
		⋮	⋮	⋮	⋮

【図 6】

絶対 X 座標	絶対 Y 座標	隣接ノード数	主要幹線道路フラグ	隣接ノード番号	リンク長
600 ⋮ ⋮ ⋮	610 ⋮ ⋮ ⋮	620 ⋮ ⋮ ⋮	630 ⋮ ⋮ ⋮	640 ⋮ ⋮ ⋮	650 ⋮ ⋮ ⋮
200	400	3	0	49	24
			0	50	32
			2	52	12
640	320	3	2	51	24
			1	53	37
			0	54	18
⋮ ⋮ ⋮	⋮ ⋮ ⋮	⋮ ⋮ ⋮	⋮ ⋮ ⋮	⋮ ⋮ ⋮	⋮ ⋮ ⋮
df[i]	dn[i][0]	df[i]	dc[i][0]	db[i][0]	dl[i][0]
			dc[i][j]	db[i][j]	dl[i][j]

51

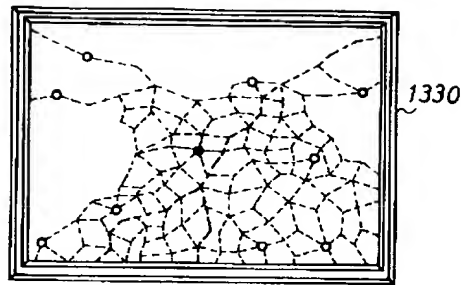
52

i

【図 9】

900 隣接ノード数	910 隣接ノード番号	920 関係リンク数
⋮	⋮	⋮
3	49	
	50	
	52	5
⋮	⋮	⋮
b n[i]	b c[i][0]	b n[i][0]
	b c[i][j]	b n[i][j]

【図 14】



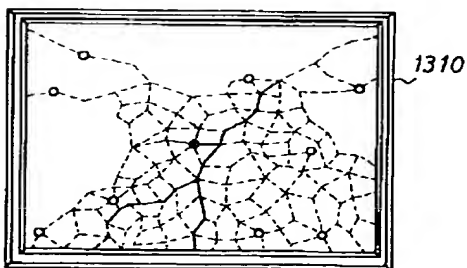
【図 10】

920 関係リンク数	930 関係リンク 始点	940 関係リンク ポイント	950 相関係数 速度←速度	960 相関係数 速度←密度	970 相関係数 密度←密度	980 相関係数 密度←速度
⋮	⋮	⋮	⋮	⋮	⋮	⋮
5	49	2	0.10	0.03	0.10	0.02
	50	1	0.05	0.01	0.05	0.01
	51	2	0.60	0.05	0.70	0.05
	52	1	0.05	0.01	0.05	0.01
	52	2	0.02	0.03	0.05	0.03
⋮	⋮	⋮	⋮	⋮	⋮	⋮
b n[i][0]	b s[i][0][0]	b p[i][0][0]	b v v[i][0][0]	b m v[i][0][0]	b v m[i][0][0]	b m m[i][0][0]
	b s[i][0][k]	b p[i][0][k]	b v v[i][0][k]	b m v[i][0][k]	b v m[i][0][k]	b m m[i][0][k]
b n[i][j]	b s[i][j][0]	b p[i][j][0]	b v v[i][j][0]	b m v[i][j][0]	b v m[i][j][0]	b m m[i][j][0]
	b s[i][j][k]	b p[i][j][k]	b v v[i][j][k]	b m v[i][j][k]	b v m[i][j][k]	b m m[i][j][k]

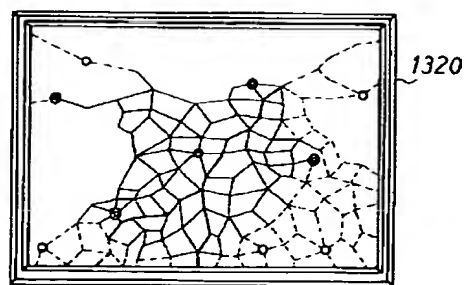
【図11】

	1000 └─┘	1010 └─┘	1020 └─┘	1030 └─┘	1040 └─┘
	絶対X座標	絶対Y座標	隣接ノード数	隣接ノード番号	リンク通過時間
	⋮	⋮	⋮	⋮	⋮
51	200	400	3	49	49
				50	50
				51	51
52	640	320	3	11	11
				11	11
				12	12
	⋮	⋮	⋮	⋮	⋮
i	$nx[i]$	$ny[i][0]$	$nn[i]$	$nc[i][0]$	$nt[i][0]$
				⋮	⋮
				$nc[i][j]$	$nt[i][j]$

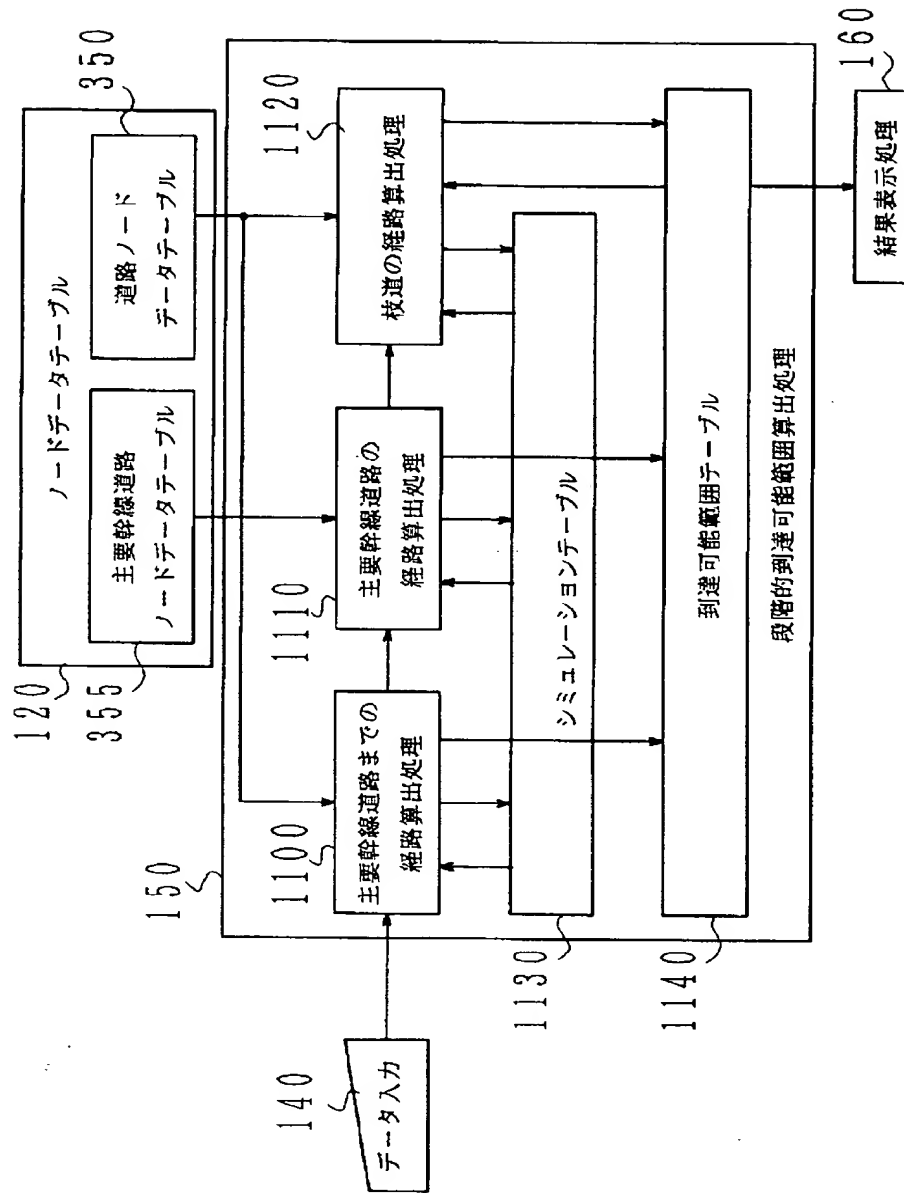
【図15】



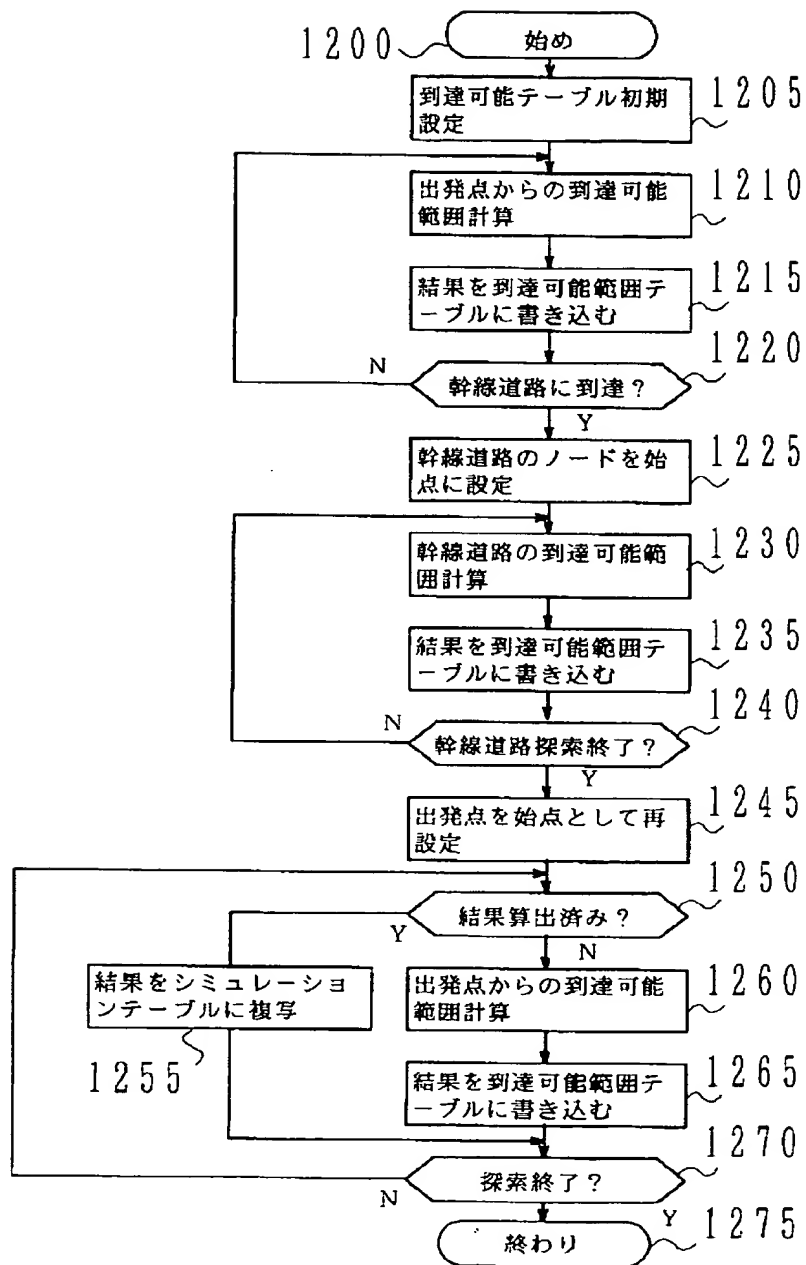
【図16】



【図12】



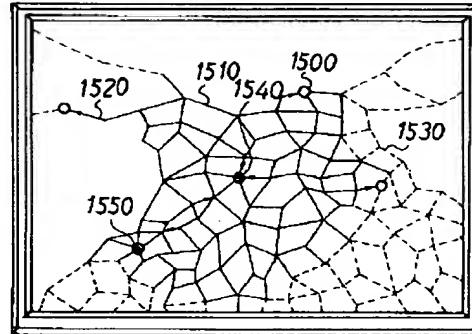
【図 1 3】



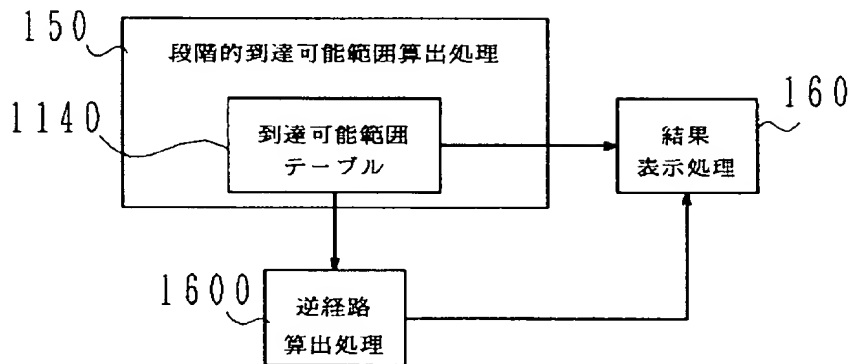
【図 17】

ノード番号	絶対X座標	絶対Y座標	到達時間	親ノード番号
48	100	300	0	48
50	250	450	5	48
⋮	⋮	⋮	⋮	⋮
61	200	400	16	50
⋮	⋮	⋮	⋮	⋮
52	640	320	30	51
⋮	⋮	⋮	⋮	⋮
nf[i]	of[i]	na[i][0]	nf[i]	of[i]

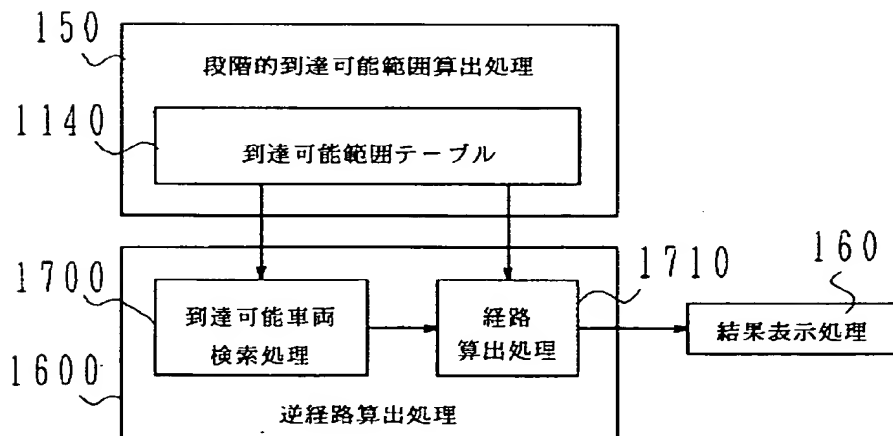
【図 18】



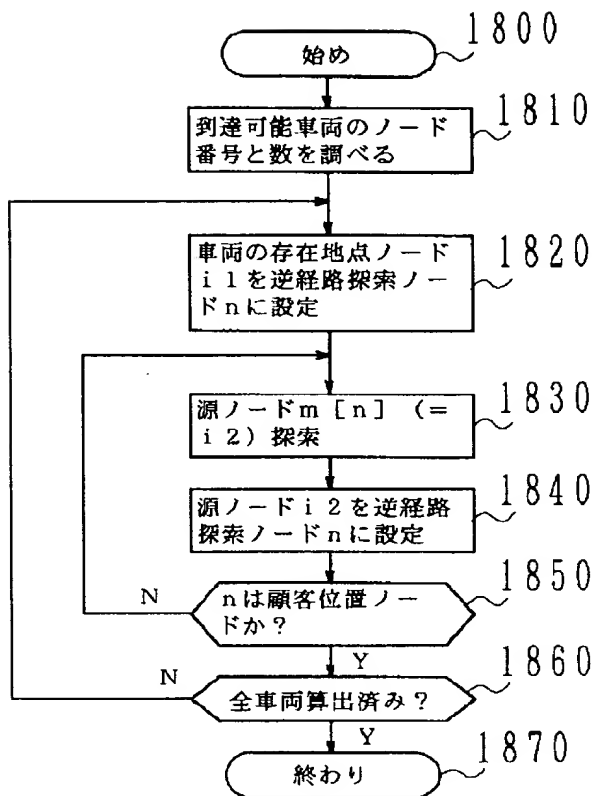
【図 19】



【図 20】



【図 21】

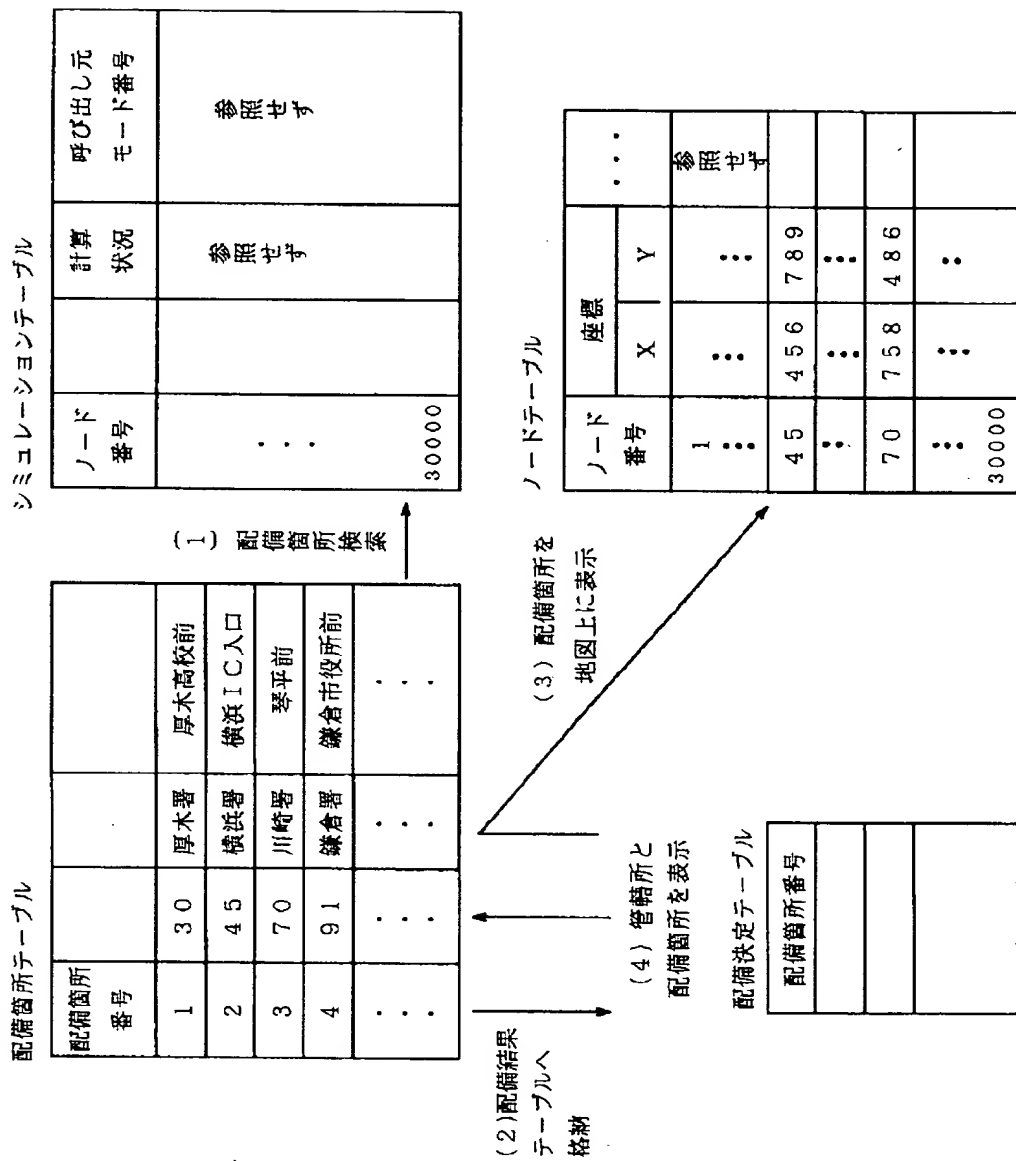


【図 23】

候補ノードテーブル

候補ノードの ノード番号	ノード 番号	到達時間	計算状況	呼び出し元 ノード番号
2	1	999999	未計算	null
8	2	8	候補	5
4	3	5	確定	5
1	4	6	候補	7
6	5	0	確定	5
⋮	6	999999	未計算	null
⋮	7	4	確定	5
⋮	8	7	候補	7
	↓	⋮	⋮	⋮
	30000	⋮	⋮	⋮

【図22】

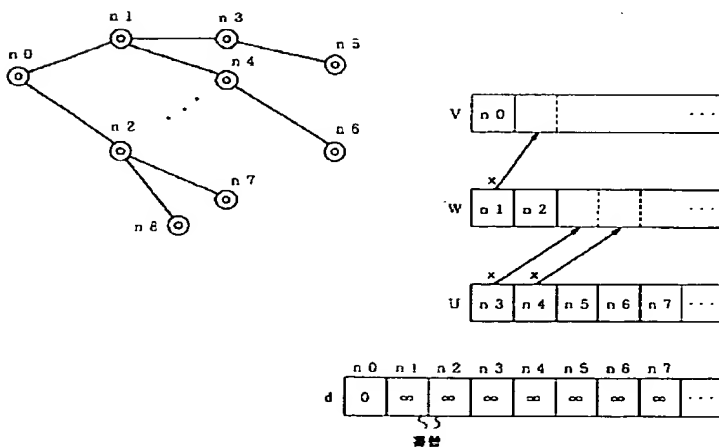


【図 24】

アドレスポイントテーブル

	シミュレーション ノード番号	ノード 番号	到達時間	計算状況	呼び出し元 ノード番号
	5	1	999999	未計算	null
	7	2	8	確定	5
リードポインタ →	3	3	5	確定	5
	2	4	6	確定	7
	8	5	0	確定	5
	4	6	999999	未計算	null
ライトポインタ →	⋮	7	4	確定	5
	⋮	8	8	確定	5
	⋮	↓	⋮	⋮	⋮
		30000	⋮	⋮	⋮

【図 27】



【図 26】

バトカーテーブル

バトカーの ノード番号
30
45
82
⋮

(1) 候補バトカーの検索

シミュレーションテーブル

ノード 番号	到達時間 (秒)	計算状況	呼び出し元 ノード番号
0	⋮	⋮	⋮
10	12	⋮	15
⋮	⋮	⋮	⋮
15	6	⋮	48
⋮	⋮	⋮	⋮
30	21	参 照 せ ず	10
⋮	⋮	⋮	⋮
45	50	⋮	20
⋮	⋮	⋮	⋮
48	0	⋮	48
⋮	⋮	⋮	⋮
82	999999	⋮	44
⋮	⋮	⋮	⋮
30000	⋮	⋮	⋮

(4) バトカー経路検索

バトカー経
路テーブル

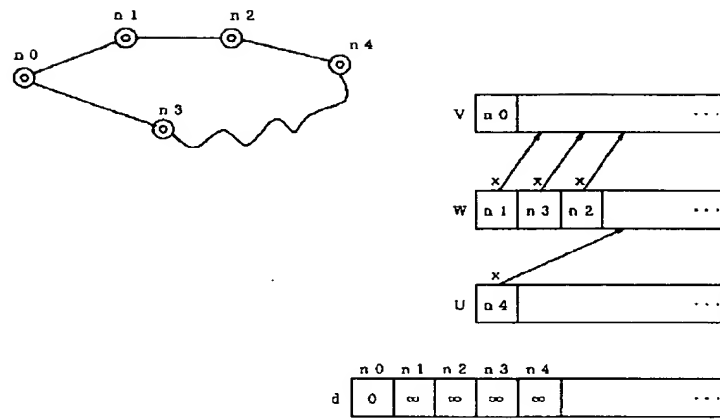
バトカーの ノード番号
30
10
15
48
⋮

(5) 急行バトカーと
バトカーの経路表示

ノードテーブル

ノード 番号	座標	
	X	Y	
0	⋮	⋮	⋮
10	545	128	⋮
⋮	⋮	⋮	⋮
15	548	159	⋮
⋮	⋮	⋮	⋮
30	512	115	参 照 せ ず
⋮	⋮	⋮	⋮
48	484	197	⋮
⋮	⋮	⋮	⋮
30000	⋮	⋮	⋮

【図 28】



フロントページの続き

(72) 発明者 北澤 修司
 東京都千代田区神田駿河台四丁目 6 番地
 株式会社日立製作所内

**This Page is Inserted by IFW Indexing and Scanning
Operations and is not part of the Official Record**

BEST AVAILABLE IMAGES

Defective images within this document are accurate representations of the original documents submitted by the applicant.

Defects in the images include but are not limited to the items checked:

- ☐ BLACK BORDERS
- ☐ IMAGE CUT OFF AT TOP, BOTTOM OR SIDES
- ☒ ~~FADED TEXT OR DRAWING~~
- ☐ BLURRED OR ILLEGIBLE TEXT OR DRAWING
- ☐ SKEWED/SLANTED IMAGES
- ☐ COLOR OR BLACK AND WHITE PHOTOGRAPHS
- ☐ ~~GRAY SCALE DOCUMENTS~~
- ☒ ~~LINES OR MARKS ON ORIGINAL DOCUMENT~~
- ☐ REFERENCE(S) OR EXHIBIT(S) SUBMITTED ARE POOR QUALITY
- ☐ OTHER: _____

IMAGES ARE BEST AVAILABLE COPY.

As rescanning these documents will not correct the image problems checked, please do not report these problems to the IFW Image Problem Mailbox.